## # Vulnerability scanning using Nikto in Kali Linux.

Nikto, also known as Nikto2, is an open source (GPL) and free-to-use web server scanner which performs vulnerability scanning against web servers for multiple items including dangerous files and programs, and checks for outdated versions of web server software. It also checks for server configuration errors and any possible vulnerabilities they might have introduced.

### Nikto installation:

The Nikto vulnerability scanner can be installed in multiple ways on both Windows- and Linux-based systems. It is available in package format on Linux for easy installation via a package manager (apt, yum, etc.) and also available via GitHub to be installed or run directly from the project source. We will be downloading it on our Virtual Kali Linux using the ‘apt’ package.

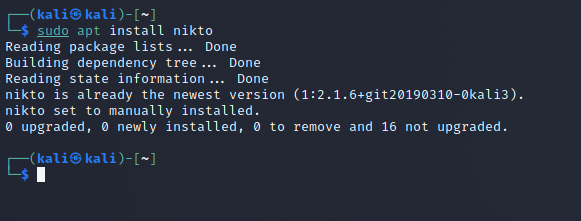
First, we will refresh our APT package lists and install any pending updates by the following command:

$ sudo apt update

**

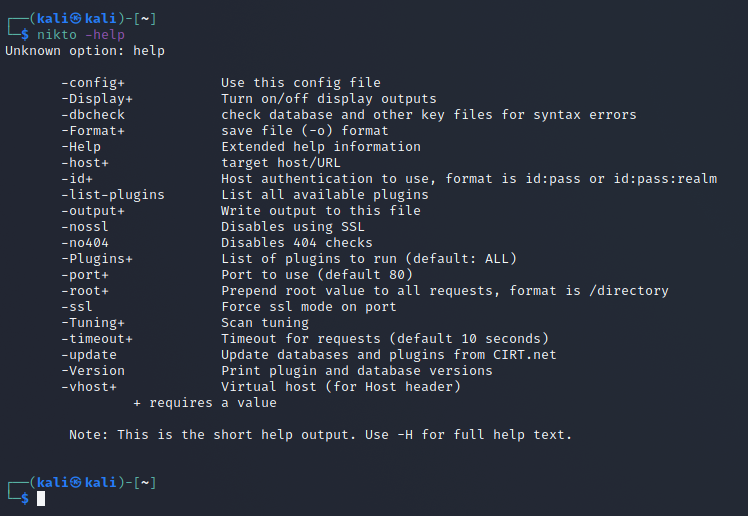
Next, we will install the Nikto web scanner with the command:

$ sudo apt install Nikto



To verify that the Nikto website vulnerability scanner is installed and ready for use, run the command to see a detailed guide on all the inputs Nikto can take and what each input does.

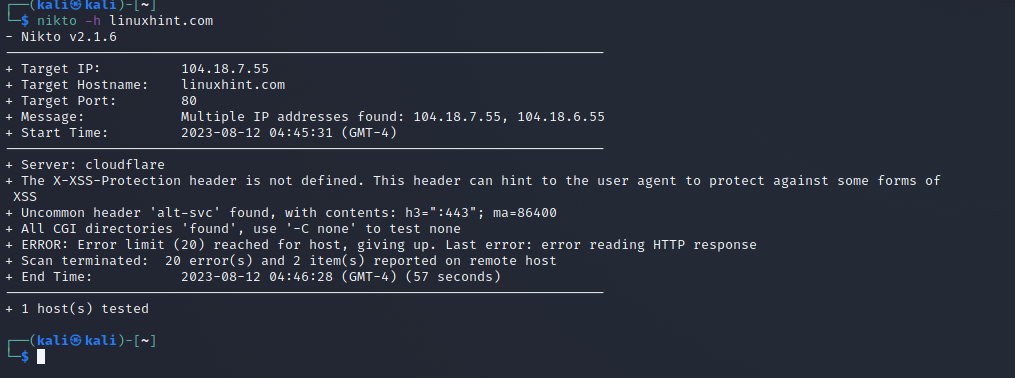
$ nikto -help



### Running a basic website scan:

The most basic way to scan a host with Nikto is to use the -h flag with the nikto command as follows:

$ nikto -h linuxhint.com

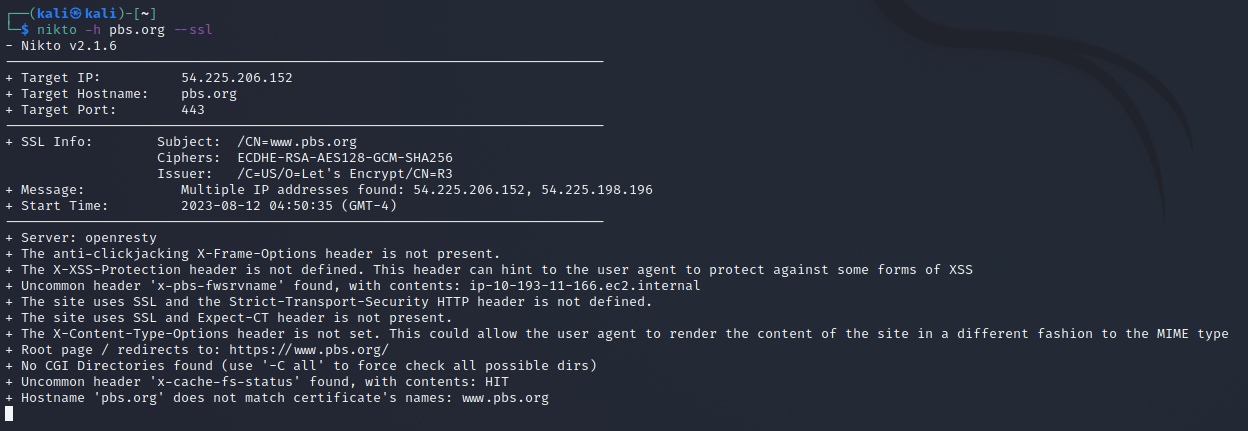


### Running a scan on a website with SSL

Nikto also has an SSL scanner mode, for SSL certificates installed on a website. We can get SSL cipher and issuer information by running the following command:

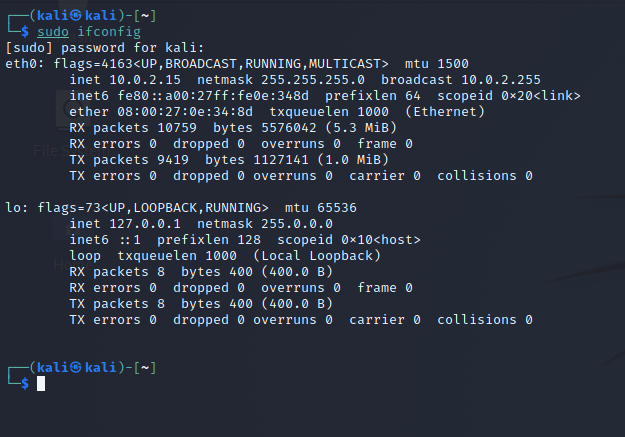
$ nikto -h pbs.org –ssl

When scanning with the -ssl option enabled, we can find more vulnerabilities and configuration errors present in the web server we've just scanned when compared to the non-ssl scan. This is often observed with misconfigured web servers, which hastily include SSL support. Thus, scanning both http and https is vital for a complete picture of the vulnerabilities present in a web server setup.



### Scanning IP addresses:

Nikto can also be used on the local network to search for any embedded servers. To do that, we’ll need to know our IP address. Type the following in command terminal:

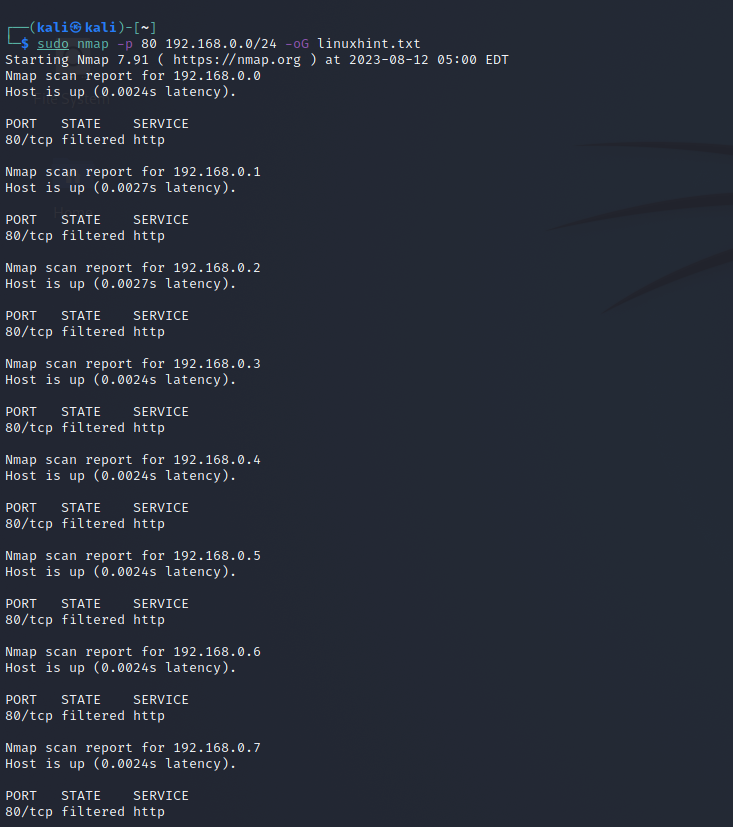
$ sudo ifconfig

### Scanning specific ports:

On certain deployments, web servers are run on non-standard ports like 8081 or 8080, or multiple web servers are run on the same host on different network ports. It's therefore vital to have the ability to scan specific ports as well as the main 80 and 443 ports.

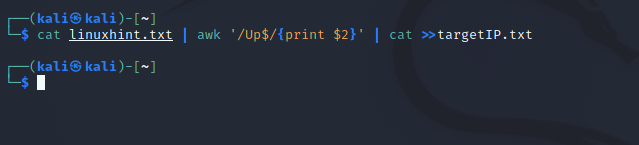
We will be scanning for services running on our local network by scanning port 80 within our range using Nmap. This will extract only the hosts that are currently up and running, as these are the ones with their port 80 open. Here, we’re exporting this data into a file we’ll name linuxhint.txt:

$ sudo nmap -p 80 192.168.0.0/24 -oG linuxhint.txt



To forward the list of all the current hosts extracted by Nmap to Nikto, we can use the cat to read the file we exported the info to. Here’s the code to run to do so:

$ cat linuxhint.txt | awk '/Up$/{print $2}' | cat >> targetIP.txt

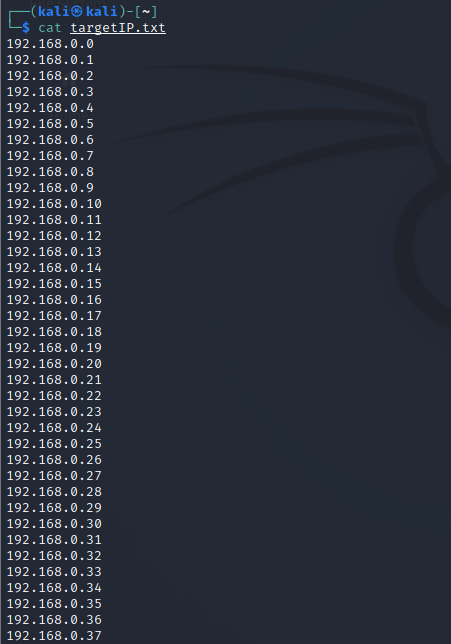


Let’s check each element in the previously stated line of code.

* **awk**– searches for the pattern within the targeted file that follows ‘cat’ in the syntax
* **Up**– about the fact that the host is up
* **Print $2**– means you’re instructing to print out the second word in each line of the .txt file
* **targetIP.txt** - is just a file we’re sending our data to, which in your case you can name it whatever you want.

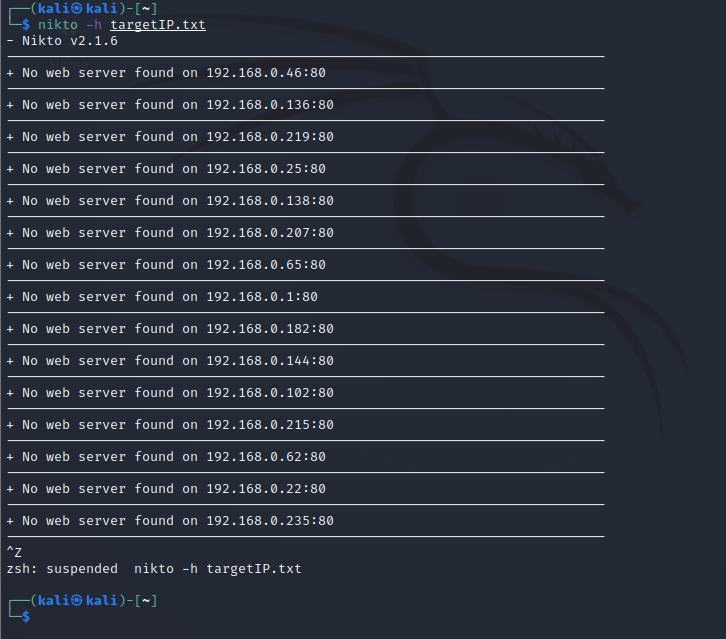
Now we can access our new file, which is targetIP in our case, to see what IP addresses have port 80 open.

$ cat targetIP.txt



Send this output file over to Nikto with the following command:

$ nikto -h targetIP.txt



### Scanning HTTP websites:

The information displayed after scanning a particular website which is most relevant to our users is the directories it has located. We can use these to acquire any user credentials, among other things that were either misconfigured or were unintentionally left open to be accessed. We can scan a website by running the following command:

$ nikto -h [www.afl.com.au](http://www.afl.com.au)

